

CLAIMS

1. (Cancelled)
2. (Currently Amended) A method according to claim [1] 17 wherein the step of operating upon each of the first threads performs lossless compression.
3. (Currently Amended) A method according to claim [1] 17 wherein the step of operating upon each of the first threads independently operates upon each of the plurality of first threads.
4. (Currently Amended) A method according to claim [1] 17 wherein at least certain ones of the first threads are independently operated upon in parallel.
5. (Cancelled)
6. (Currently Amended) A method according to claim [1] 17 further comprising the step of combining compressed blocks in each of the plurality of compressed first threads to obtain digitally compressed data.
7. (Currently Amended) A method according to claim [1] 17 wherein the step of creating the plurality of first threads includes the step of associating each of the plurality of blocks of digital data with one of the plurality of first threads such that blocks within each of the plurality of first threads share certain common compression characteristics.
8. – 10. (Cancelled)
11. (Currently Amended) A method according to claim [1] 17 wherein the step of creating each of the plurality of first threads uses a data type of each of the plurality of blocks so that each of the first threads contains blocks which have a similar data type.

12. – 13. (Cancelled)

14. (Currently Amended) A method according to claim [1] 17 further including the step of predicting an estimated compression time and estimated compression amount for each block.

15. (Cancelled)

16. (Currently Amended) A method according to claim [1] 17 wherein the step of partitioning data includes the step of determining a size of each of the plurality of blocks taking data type of each block into account.

17. (Currently Amended) ~~A method according to claim 1 further including the steps of:~~
A method of operating upon digital data comprising the steps of:

partitioning the digital data into a plurality of blocks;

creating a plurality of first threads, such that each first thread includes at least one of the plurality of blocks; and

operating upon each of the plurality of first threads to obtain a plurality of compressed first threads, each compressed first thread including at least one compressed block of digital data;

operating upon each of the compressed first threads to eliminate each of the compressed first threads and retain the compressed first blocks;

creating a plurality of second threads, such that each second thread includes at least one of the plurality of compressed first blocks; and

operating upon each of the plurality of second threads to obtain a plurality of compressed second threads, each compressed second thread including at least one compressed second block of digital data.

18. (Original) A method according to claim 17 wherein the step of operating upon each of the second threads independently operates upon each of the plurality of second threads.

19. (Original) A method according to claim 17 wherein at least certain ones of the second threads are independently operated upon in parallel.

20. (Original) A method according to claim 17 wherein, during the step of operating upon each of the plurality of second threads, the same compression algorithm used to operate upon each block is also used to operate upon the corresponding compressed block.

21. (Previously Presented) A method according to claim 17 further comprising the step of combining the compressed blocks in each of the plurality of compressed second threads to obtain digitally compressed data.

22. (Original) A method according to claim 17 wherein the step of creating the plurality of second threads includes the step of associating each of the plurality of compressed first blocks with one of the plurality of second threads such that compressed first blocks within each of the plurality of second threads share certain common compression characteristics.

23. – 25. (Cancelled)

26. (Original) A method according to claim 17 wherein the step of operating upon each of the plurality of first threads also results in obtaining a plurality of first metadata sets, each first metadata set including portions of compressed first blocks which are determined to possibly have redundancies disposed therein.

27. – 28. (Cancelled)

29. (Original) A method according to claim 17 wherein each first thread has an associated first metadata set.

30. – 46. (Cancelled)

47. (Currently Amended) A method according to claim [1] 17 wherein each first thread further includes control signals.

48. – 53. (Cancelled)

54. (Previously Presented) A method according to claim 19 wherein at least certain ones of the first threads are independently operated upon in parallel.

55-63. (Cancelled)

64. (Previously Presented) A method according to claim 61, wherein the step of compressing further includes the steps of:

operating upon each of the compressed first threads to eliminate each of the compressed first threads and retain the compressed first blocks;

creating a plurality of second threads, such that each second thread includes at least one of the plurality of compressed first blocks; and

operating upon each of the plurality of second threads to obtain a plurality of compressed second threads, each compressed second thread including at least one compressed second block of digital data.

65. (Previously Presented) A method according to claim 64 wherein at least certain ones of the second threads are independently operated upon in parallel.

66. (Previously Presented) A method according to claim 64 wherein, during the step of operating upon each of the plurality of second threads, the same compression algorithm used to operate upon each block is also used to operate upon the corresponding compressed block.

67 – 73. (Cancelled)